

WSPRPi: Reprogramming PIC32 from Raspberry Pi Zero

Dan McGraw M0WUT

Abstract

This document details how to setup the Pi Zero in WSPRpi to allow it to reprogram the PIC32 for automatic firmware updates. It assumes that the UART bootloader has been installed on the PIC32. (See "Installing Bootloader" in the Documentation section for more info)

Anything written in this font is instructional text.

Anything written in this font should be typed in exactly into the terminal on the Pi. Each item starting with a \$ is a separate command.

- Make sure we're in the user directory:

```
$ cd
```

- Git clone Pic32prog utility:

```
$ git clone https://github.com/majenkotech/pic32prog-autotools.git
```

- Install gettext and autoconf:

```
$ sudo apt install gettext autoconf
```

- Navigate to pic32prog-autotools directory:

```
$ cd pic32prog-autotools
```

- Install (these instructions take a while and some don't print anything so be patient)

```
$ autoreconf -fi  
$ ./configure  
$ make
```

- Put the WSPRpi into bootloader mode

While holding the "Menu" button on the front panel, tap the "Reset" button on the main PCB. The front panel LED should blink rapidly.

- Download a test hex file Change directory back to your home directory and download a test hex file.

```
$ cd  
$ wget https://github.com/WSPRpi/PIC32-Firmware/blob/master/Bootloader/test.hex
```

- Program the PIC

```
$ pic32prog -d /dev/ttyAMA0 -b 115200 test.hex
```

Hopefully, the result will look something like shown below. Note that I hadn't written the code for test.hex yet so I used WSPR_TCVR.ino.hex. This should be replaced with the name of the hex file you are uploading. Note that the number of # symbols are an indication of program size so you may have more or less depending on the complexity of the file you're uploading.

```

pi@WSPRpi:~ $ pic32prog -d /dev/ttyAMA0 -b 115200 WSPR_TCVR.ino.hex
Copyright: (C) 2011-2015 Serge Vakulenko
          2016-2017 Majenko Technologies
Programmer for Microchip PIC32 microcontrollers, Version 2.1.36
Adapter: STK500v2 Bootloader
Program area: 1d000000-1dffffff
Processor: Bootloader
Flash memory: 2048 kbytes
Boot memory: 80 kbytes
Data: 100940 bytes
Erase: done
Program flash: ##### done
Verify flash: ##### done
Program rate: 4671 bytes per second
pi@WSPRpi:~ $

```

Figure 1: Successful upload of WSPR_TCVR.ino.hex

If you haven't put the PIC into bootloader mode or there is a communication error, you will probably get a screen as below:

```

pi@WSPRpi:~ $ pic32prog -d /dev/ttyAMA0 -b 115200 WSPR_TCVR.ino.hex
Copyright: (C) 2011-2015 Serge Vakulenko
          2016-2017 Majenko Technologies
Programmer for Microchip PIC32 microcontrollers, Version 2.1.36

No target found.
pi@WSPRpi:~ $

```

Figure 2: Failed upload of WSPR_TCVR.ino.hex

If this is the case, check that the PIC was in bootloader mode (LED blinking rapidly), other than that, it is probably poor solder joints.

- Admire the pretty lights

If everything has gone well, the front panel LED (and the TX LED on the main PCB) should be repeatedly flashing MOWUT in Morse.

This is - - / - - - - - / . - - / .. - / -